

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
 - TEXT CUT OFF AT TOP, BOTTOM OR SIDES
 - FADED TEXT
 - ILLEGIBLE TEXT
 - SKEWED/SLANTED IMAGES
 - COLORED PHOTOS
 - BLACK OR VERY BLACK AND WHITE DARK PHOTOS
 - GRAY SCALE DOCUMENTS
-

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**



Subscribe Register Login
(Full Service) (Limited Service, Free)

Search: ☒ The ACM Digital Library ☐ The Guide
invokeInterface <and> index

THE ACM DIGITAL LIBRARY

Feedback

Terms used invokeInterface and index

Sort results
by

relevance



Display results

expanded form



☒ Save results to a Binder

Try

☒ Search Tips

Try

☐ Open results in a new window

Results 1 - 20 of 200

Result page: **1** 2 3 4 5 6 7 8 9

Best 200 shown

1 Dynamic metrics for java

Bruno Dufour, Karel Driesen, Laurie Hendren, Clark Verbrugge

October 2003 ACM SIGPLAN Notices , Proceedings of the 18th ACM SIGPLAN conference on programming languages, systems, languages, and applications, Volume 38 Issue 11

Full text available: pdf(222.67 KB)

Additional Information: full citation, abstract, references,

In order to perform meaningful experiments in optimizing compilation and run time analysis, researchers usually rely on a suite of benchmark programs of interest to the optimization community. Programs are described as *numeric*, *memory-intensive*, *concurrent*, or *object-oriented*, in some cases with little justification. We believe it is beneficial to have a concise and precisely ...

Keywords: Java, dynamic metrics, execution traces, optimization, profiling, pr

2 The hB-tree: a multiattribute indexing method with good guaranteed performance





David B. Lomet, Betty Salzberg

December 1990 ACM Transactions on Database Systems (TODS), Volume 15

Full text available: pdf(2.58 MB)

Additional Information: full citation, abstract, references, citing


A new multiattribute index structure called the hB-tree is introduced. It is derived from the B-tree [15] but has additional desirable properties. The hB-tree internode search and intranode search are analogous to the corresponding processes in B-trees [1]. The intranode process involves traversing the structure within nodes for very efficient searching. Node splitting requires nodes which no longer represent ...

- 3 **Query processing: Factorizing complex predicates in queries to exploit inde**
Surajit Chaudhuri, Prasanna Ganesan, Sunita Sarawagi
June 2003 Proceedings of the 2003 ACM SIGMOD international conference on M
Full text available:  pdf(240.56 KB) Additional Information: full citation, abstract, reference
Decision-support applications generate queries with complex predicates. We s
query expressions exposes significant opportunities for exploiting available inc
relaxing predicates in a complex condition to create possibilities for factoring.
integration with existing query optimizers and support multiple optimization le
between plan complexity and ...
- 4 **XML indexing and compression: D(k)-index: an adaptive structural summar**
Qun Chen, Andrew Lim, Kian Win Ong
June 2003 Proceedings of the 2003 ACM SIGMOD international conference on M
Full text available:  pdf(217.30 KB) Additional Information: full citation, abstract, referen
To facilitate queries over semi-structured data, various structural summaries l
summaries are derived directly from the data and serve as indices for evaluat
or XML data. We introduce the D(k) index, an adaptive structural summary for
Building on previous work, 1-index and A(k) index, the D(k)-index is also bas
However, as a generalization of the 1-index and A ...
- 5 **XML indexing and compression: ViST: a dynamic index method for queryin**
Haixun Wang, Sanghyun Park, Wei Fan, Philip S. Yu
June 2003 Proceedings of the 2003 ACM SIGMOD international conference on M
Full text available:  pdf(244.47 KB) Additional Information: full citation, abstract, references,
With the growing importance of XML in data exchange, much research has bee
facilities to extract data from structured XML documents. In this paper, we pr
searching XML documents. By representing both XML documents and XML que
show that querying XML data is equivalent to finding subsequence matches. U
query into multiple sub-queries, and then *join* ...
- 6 **Surrogate subsets: a free space management strategy for the index of a te**
F. J. Burkowski
December 1989 Proceedings of the 13th annual international ACM SIGIR conferenc
information retrieval
Full text available:  pdf(1.44 MB) Additional Information: full citation, abstract, references, ci
This paper presents a new data structure and an associated strategy to be util
retrieval systems. The paper starts by reviewing some of the goals that may b
index and continues with a small survey of various current strategies. It then
to as surrogate subsets discussing its appropriateness in the light of the speci
implementation details are disc ...

11 Session 12C: High-order entropy-compressed text indexes

Roberto Grossi, Ankur Gupta, Jeffrey Scott Vitter

January 2003 Proceedings of the fourteenth annual ACM-SIAM symposium on D

Full text available:  pdf(1.14 MB)

Additional Information: full citation, abstract, references, citir

We present a novel implementation of compressed suffix arrays exhibiting new space occupancy for a given text (or sequence) of n symbols over an alphabet encoded by $\lg \sigma$ bits. We show that compressed suffix while retaining full text indexing functionalities, such as searching any pattern ...

12 The automatic indexing system AIR/PHYS - from research to applications

P. Biebricher, N. Fuhr, G. Lustig, M. Schwantner, G. Knorz

May 1988 Proceedings of the 11th annual international ACM SIGIR conference on information retrieval

Full text available:  pdf(785.93 KB)


Additional Information: full citation, abstract, references,

Since October 1985, the automatic indexing system AIR/PHYS has been used data base of the Fachinformationszentrum Karlsruhe/West Germany. The texts English. The system of descriptors is prescribed. For the application of the AIR containing more than 600 000 word-descriptor relations reap. phrase-descript these relations have been obtained ...

13 Special system-oriented section: the best of SIGMOD '94: Estimating page LRU buffers

Arun Swami, K. Bernhard Schiefer

October 1995 The VLDB Journal &mdash; The International Journal on Very Large

Full text available:  pdf(1.04 MB)

Additional Information: full citation, abstract, re

We describe an algorithm for estimating the number of page fetches for a part The algorithm obtains estimates for the number of page fetches for an index s selected and the number of LRU buffers currently available. The algorithm has exactly once before any estimates are calculated. This initial phase, involving all the index entries and calculates ...

Keywords: LRU, estimation, index scan, query optimization

17 Performance comparison of property map and bitmap indexing

Ashima Gupta, Karen C. Davis, Jennifer Grommon-Litton

November 2002 Proceedings of the 5th ACM international workshop on Data Warehousing

Full text available:  pdf(250.60 KB)

Additional Information: full citation, abstract, references


A data warehouse is a collection of data from different sources that supports a query and allows fast access to individual attribute values that are needed to answer a query. A Property Map (PMap) is a map that pre-computes attribute expressions, called properties, for each tuple and stores them in a bit string [LD02]. This paper compares the performance of PMap and bitmap indexing.

Keywords: bitmap index, data warehouse, performance study

18 Physical database design: Applying approximate order dependency to reduce indexing space

Jirun Dong, Richard Hull

June 1982 Proceedings of the 1982 ACM SIGMOD international conference on Management of data

Full text available:  pdf(1.22 MB)

Additional Information: full citation, abstract, references

The recently introduced notion of order dependency in the relational model is generalized. It is shown that order dependency is satisfied in an approximate way. Two fundamental types of order dependency are distinguished and analyzed. It is shown for both types that such approximate order dependency can be applied to substantially reduce indexing space without significantly increasing query response time.

19 Regular contributions: An improved index function for (D)FCM predictors

Martin Burtscher

June 2002 ACM SIGARCH Computer Architecture News, Volume 30 Issue 3

Full text available:  pdf(619.97 KB)

Additional Information: full citation, abstract, references

The most promising value predictors to date are the finite context method predictor and the differential finite context method predictor. Both predictors compute the second level value as a function of the content of the first level. This index function is improved by a new function. However, our research shows that the currently used select-fold-shift-xor function is not optimal. Sequences of values. For example, it does ...

20 Optimal indexing using near-minimal space

C. Heeren, H. V. Jagadish, L. Pitt

June 2003 Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Database Management Systems

Full text available:  pdf(208.43 KB)

Additional Information: full citation, abstract, references

We consider the index selection problem. Given either a fixed query workload or a set of possible future queries, and a bound B on how much space is available to build a collection of indices for which the average query response time is minimized. We show that the performance bounds. Let m be the number of queries in the workload. We show that the collection of indices using space ...

Compact Dispatch Tables for Dynamically Typed Object Oriented Languages [\(Make Corrections\)](#) [\(11 citations\)](#)

Jan Vitek, R. Nigel Horspool

International Conference on Compiler Construction (CC 1996)

View or download:

csr.uvic.ca/~nigelh/Publica...cdt95.pdf

Cached: [PS.gz](#) [PS](#) [PDF](#) [Image](#) [Update](#) [Help](#)



[Home/Search](#) [Bookmark](#) [Context](#) [Related](#)

From: csr.uvic.ca/~nigelh/pubs [\(more\)](#)

[\(Enter author homepages\)](#)

[\(Enter summary\)](#)

Rate this article: 1 2 3 4 5 (best)

[Comment on this article](#)

Abstract: Dynamically typed object-oriented languages must perform dynamic binding for most message sends. Typically this is slow. A number of papers have reported on attempts to adapt C++-style selector table indexing to dynamically typed languages, but it is difficult to generate space-efficient tables. Our algorithm generates considerably smaller dispatch tables for languages with single inheritance than its predecessors at the cost of a small dispatch time penalty. [\(Update\)](#)

Context of citations to this paper: [More](#)

...entries need such a test. **Subtype tests are implemented with a simple series of logical operations (a bit wise AND and a comparison)** [129]. Figure 17 shows the code for a call through a CT dispatch table. This version of the algorithm (from [128] only handles single...

...solution has $O(1)$ query time. **Unfortunately it uses $O(nM)$ table space, which is prohibitively large for existing hierarchies** [D93, M95, V95, FM96, MM96]. To cope with this, all known practical solutions use the observation that many queries return NIL. Thus they employ various...

Cited by: [More](#)

Efficient Implementation of Java Interfaces.. - Alpern, Cocchi.. (2001) [\(Correct\)](#)

Efficient Type Inclusion Tests - Vitek, Horspool, Krall (1997) [\(Correct\)](#)

Multi-Method Dispatching: A Geometric Approach with.. - Ferragina.. (1999) [\(Correct\)](#)

Similar documents (at the sentence level):

25.1%: Compact Dispatch Tables for Dynamically Typed Programming Languages - Vitek (1995) [\(Correct\)](#)

Active bibliography (related documents): [More](#) [All](#)

0.1: A General Framework For Inheritance Management and Method.. - Holst, Szafron (1997) [\(Correct\)](#)

0.1: Message Dispatch on Pipelined Processors - Driesen, Hölzle, Vitek (1995) [\(Correct\)](#)

0.1: Software and Hardware Techniques for Efficient Polymorphic Calls - Driesen (1999) [\(Correct\)](#)

Similar documents based on text: [More](#) [All](#)

0.7: Taming Message Passing: Efficient Method Look-Up for.. - Vitek, Horspool (1994) [\(Correct\)](#)

0.4: Selector Table Indexing Sparse Arrays - Driesen (1993) [\(Correct\)](#)

0.3: Iterative Type Analysis and Extended Message Splitting.. - Chambers, Ungar (1990) [\(Correct\)](#)

Related documents from co-citation: [More](#) [All](#)

8: Method Lookup Strategies in Dynamically Typed Object-Oriented Programming Langua.. (context) - Driesen - 1993

8: Selector Table Indexing and Sparse Arrays (context) - Driesen - 1993

8: Technical Brief (context) - Inc - 1992

BibTeX entry: [\(Update\)](#)

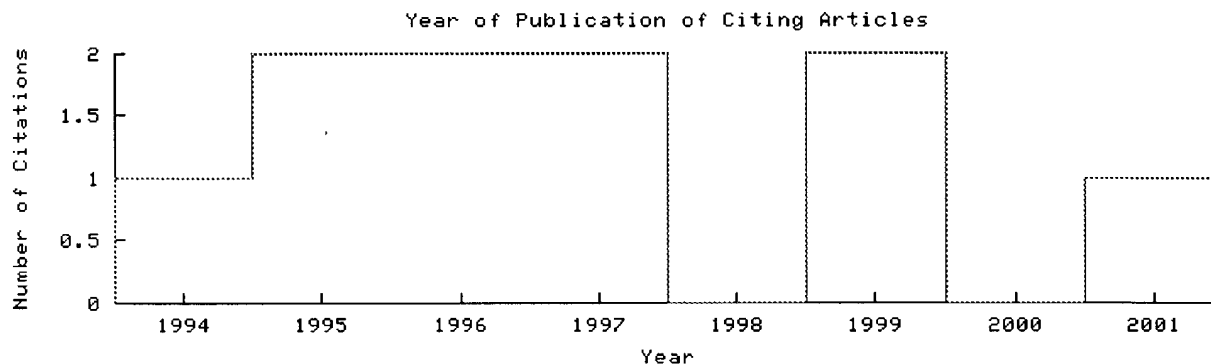
Jan Vitek. Compact Dispatch Tables for Dynamically-Typed Object-Oriented Languages. M.S. Thesis, University of Victoria, B.C., forthcoming. <http://citeseer.ist.psu.edu/573184.html> [More](#)

```
@inproceedings{vitek96compact,
  author = "Jan Vitek and R. Nigel Horspool",
  title = "Compact Dispatch Tables for Dynamically Typed Object Oriented Languages",
  booktitle = "International Conference on Compiler Construction (CC 1996)",
  address = "Linköping, Sweden",
  month = "April",
  publisher = "Springer",
  series = "Lecture Notes in Computer Science (LNCS), Vol. 1060",
  pages = "309-325",
  year = "1996",
```


url = "citeseer.ist.psu.edu/573184.html")

Citations (may not include all citations):

- 132 Precise Concrete Type Inference for Object-Oriented Language.. - Plevyak, Chien - 1994
- 80 The Design and Evaluation of a High Performance Smalltalk Sy.. (context) - Ungar - 1987
- 28 Efficient Method Dispatch in PCL - Kiczales, Rodriguez - 1990
- 28 Compile-time analysis of object-oriented programs (context) - Vitek, Horspool et al. - 1992
- 22 Optimizing Dynamically-Typed Object-Oriented Languages With .. (context) - Hitzle, Chambers et al. - 1993
- 14 Taming Message Passing: Efficient method lookup for dynamica.. - Vitek, Horspool - 1994
- 9 Compact Dispatch Tables for Dynamically Typed Programming La.. - Vitek - 1995
- 1 Fast Constant Time Type Inclusion Testing (context) - Vitek, Horspool - 1995



The graph only includes citing articles where the year of publication is known.

Documents on the same site (<http://www.csr.uvic.ca/~nigelh/pubs.html>): [More](#)

Incremental Generation of LR Parsers - Horspool (1989) ([Correct](#))

Faster Generalized LR Parsing - Aycock, Horspool (1999) ([Correct](#))

JAZZ: An Efficient Compressed Format for Java Archive Files - Bradley, Horspool, Vitek (1998) ([Correct](#))

[Online articles have much greater impact](#) [More about CiteSeer](#) [Add search form to your site](#) [Submit documents](#)
[Feedback](#)

CiteSeer - Copyright [NEC](#) and [IST](#)

Selector Table Indexing Sparse Arrays (1993) ([Make Corrections](#)) ([5 citations](#))Karel Driesen
Conference on Object-Oriented

View or download:

ucsb.edu/oocsb/papers/oopsla93.psCached: [PS.gz](#) [PS](#) [PDF](#) [Image](#) [Update](#) [Help](#)[Home/Search](#) [Bookmark](#) [Context](#) [Related](#)From: ucsb.edu/oocsb/papers/oopsla93 ([more](#))
([Enter author homepages](#))[\(Enter summary\)](#)

Rate this article: 1 2 3 4 5 (best)

[Comment on this article](#)

Abstract: Selector table indexing is a simple technique for method lookup in object-oriented languages, which yields good performance, is well suited to multiple inheritance and dynamic typing, but is generally disregarded for its prohibitive memory consumption. The large memory footprint is caused by keeping a table of methods, indexed by a selectorcode, for each class in the system. These tables are sparsely filled. A sparse array implementation is presented, which reduces the memory consumption by an... ([Update](#))

Context of citations to this paper: [More](#)

...800 classes and 5,000 message selectors consume 16MBytes. A number of approaches have been tried to reduce the size of these tables [2][10][9] 17] 22] The most effective of these techniques [10] achieves an average dispatch table compression of 86 ; in the above example...

.... selector table indexing has been discussed and useful optimizations presented in recent papers by Dixon et al. and by Driesen [Dix89][Drie93]. In more dynamic languages this constant time lookup would be provided at the expense of recomputing closures whenever the class...

Cited by: [More](#)Compact Dispatch Tables for Dynamically Typed Programming Languages - Vitek (1996) ([Correct](#))Efficient Implementation of Java Interfaces.. - Alpern, Cocchi.. (2001) ([Correct](#))Ideal Hash Trees - Bagwell ([Correct](#))Active bibliography (related documents): [More](#) [All](#)0.5: Compressing Sparse Tables using a Genetic Algorithm - Driesen (1994) ([Correct](#))0.5: The OBJVLISP Model: Definition of a Uniform, Reflexive.. - Jean-Pierre Briot Pierre ([Correct](#))0.3: Pushpin Computing: a Platform for Distributed Sensor Networks - Physics And Mathematics ([Correct](#))Similar documents based on text: [More](#) [All](#)0.4: Compact Dispatch Tables for Dynamically Typed Object Oriented .. - Vitek, Horspool ([Correct](#))0.2: Software and Hardware Techniques for Efficient Polymorphic Calls - Driesen (1999) ([Correct](#))0.2: Tags for High Performance Active Networks - Wolf, Decasper, Tschudin (2000) ([Correct](#))-----Related documents from co-citation: [More](#) [All](#)-----

4: A fast method dispatcher for compiled languages with multiple inheritance (context) - Dixon, Mckee et al. - 1989

3: Object-Oriented Programming: An Evolutionary Approach (context) - Cox - 1987

3: Optimizing Method Search with Lookup Caches and Incremental Coloring (context) - Andre, Royer - 1992

BibTeX entry: ([Update](#))

Driesen, K.: Selector Table Indexing & Sparse Arrays.Proc. OOPSLA'93, Washington, DC, 1993.

<http://citeseer.ist.psu.edu/driesen93selector.html> [More](#)

```
@inproceedings{ driesen93selector,
  author = "Karel Driesen",
  title = "Selector Table Indexing Sparse Arrays",
  booktitle = "Conference on Object-Oriented",
  pages = "259-270",
  year = "1993",
  url = "citeseer.ist.psu.edu/driesen93selector.html" }
```

Citations (may not include all citations):

124 Object Oriented Programming: An Evolutionary Approach (context) - Cox - 1987

38 A fast Method Dispatcher for Compiled Languages with Multipl.. (context) - Dixon, Vaughan et al.

28 Robson Smalltalk-80: The Language and its Implementation Add.. (context) - Goldberg - 1983

23 Optimizing Method Search with Lookup Caches and Incremental .. (context) - Andre, Royer

- 3 Discrete Mathematics with Algorithms (context) - Albertson, Hutchinson - 1988
- 2 An Efficient Implementation of Trie Structures Software-Prac.. (context) - Aoe, Morimoto et al. - 1992
- 2 Words of Advice (context) - Smalltalk-, History - 1983
- 1 Mixin-based inheritance ECOOP/COPSLA'90 Proceedings (context) - Bracha, Cook
- 1 Van Limberghen AGORA (context) - Codenie, Steyaert - 1991
- 1 Reference Manual AddisonWesley (context) - Ellis, The - 1990
- 1 Van Limberghen Nested Mixin-Methods in Agora to be presented.. (context) - Steyaert, Codenie et al.
- 1 Steyaert The Design and Implementation of the Agora Environm. (context) - D'Hondt - 1993
- 1 Faculty of Sciences (context) - Driesen, Smalltalk- - 1987
- 1 Efficient Implementation of the Smalltalk-80 System POPL Pro.. (context) - Deutch, Schiffman - 1984

[Online articles have much greater impact](#) [More about CiteSeer](#) [Add search form to your site](#) [Submit documents](#)
[Feedback](#)

CiteSeer - Copyright NEC and IST

Compact Dispatch Tables for Dynamically Typed Object Oriented Languages ([Make Corrections](#)) ([11 citations](#))

Jan Vitek, R. Nigel Horspool

International Conference on Compiler Construction (CC 1996)

View or download:

csr.uvic.ca/~nigelh/Publica...cdt95.pdfCached: [PS.gz](#) [PS](#) [PDF](#) [Image](#) [Update](#) [Help](#)[Home/Search](#) [Bookmark](#) [Context](#) [Related](#)From: csr.uvic.ca/~nigelh/pubs ([more](#))[\(Enter author homepages\)](#)[\(Enter summary\)](#)

Rate this article: 1 2 3 4 5 (best)

[Comment on this article](#)

Abstract: Dynamically typed object-oriented languages must perform dynamic binding for most message sends. Typically this is slow. A number of papers have reported on attempts to adapt C++-style selector table indexing to dynamically typed languages, but it is difficult to generate space-efficient tables. Our algorithm generates considerably smaller dispatch tables for languages with single inheritance than its predecessors at the cost of a small dispatch time penalty. ([Update](#))

Context of citations to this paper: [More](#)

...entries need such a test. **Subtype tests are implemented with a simple series of logical operations (a bit wise AND and a comparison)** [129]. Figure 17 shows the code for a call through a CT dispatch table. This version of the algorithm (from [128] only handles single...

...solution has $O(1)$ query time. **Unfortunately it uses $O(nM)$ table space, which is prohibitively large for existing hierarchies** [D93, M95, V95, FM96, MM96]. To cope with this, all known practical solutions use the observation that many queries return NIL. Thus they employ various...

Cited by: [More](#)Efficient Implementation of Java Interfaces.. - Alpern, Cocchi.. (2001) ([Correct](#))Efficient Type Inclusion Tests - Vitek, Horspool, Krall (1997) ([Correct](#))Multi-Method Dispatching: A Geometric Approach with.. - Ferragina.. (1999) ([Correct](#))

Similar documents (at the sentence level):

25.1%: Compact Dispatch Tables for Dynamically Typed Programming Languages - Vitek (1995) ([Correct](#))Active bibliography (related documents): [More](#) [All](#)**0.1:** A General Framework For Inheritance Management and Method.. - Holst, Szafron (1997) ([Correct](#))**0.1:** Message Dispatch on Pipelined Processors - Driesen, Hölzle, Vitek (1995) ([Correct](#))**0.1:** Software and Hardware Techniques for Efficient Polymorphic Calls - Driesen (1999) ([Correct](#))Similar documents based on text: [More](#) [All](#)**0.7:** Taming Message Passing: Efficient Method Look-Up for.. - Vitek, Horspool (1994) ([Correct](#))**0.4:** Selector Table Indexing Sparse Arrays - Driesen (1993) ([Correct](#))**0.3:** Iterative Type Analysis and Extended Message Splitting.. - Chambers, Ungar (1990) ([Correct](#))Related documents from co-citation: [More](#) [All](#)**8:** Method Lookup Strategies in Dynamically Typed Object-Oriented Programming Langua.. (context) - Driesen - 1993**8:** Selector Table Indexing and Sparse Arrays (context) - Driesen - 1993**8:** Technical Brief (context) - Inc - 1992BibTeX entry: ([Update](#))

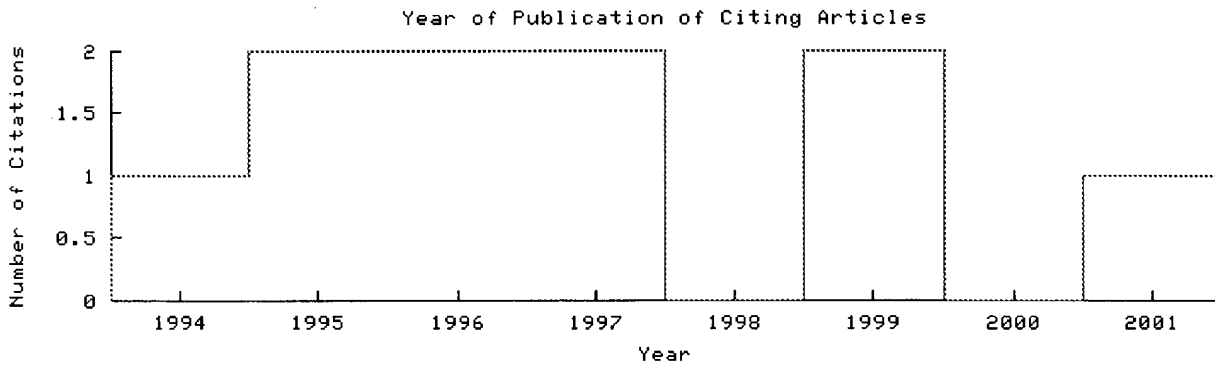
Jan Vitek. Compact Dispatch Tables for Dynamically-Typed Object-Oriented Languages. M.S. Thesis, University of Victoria, B.C., forthcoming. <http://citeseer.ist.psu.edu/573184.html> [More](#)

```
@inproceedings{vitek96compact,
  author = "Jan Vitek and R. Nigel Horspool",
  title = "Compact Dispatch Tables for Dynamically Typed Object Oriented Languages",
  booktitle = "International Conference on Compiler Construction (CC 1996)",
  address = "Linkoping, Sweden",
  month = "April",
  publisher = "Springer",
  series = "Lecture Notes in Computer Science (LNCS), Vol. 1060",
  pages = "309-325",
  year = "1996",
```

url = "citeseer.ist.psu.edu/573184.html")

Citations (may not include all citations):

- 132 Precise Concrete Type Inference for Object-Oriented Language.. - Plevyak, Chien - 1994
- 80 The Design and Evaluation of a High Performance Smalltalk Sy.. (context) - Ungar - 1987
- 28 Efficient Method Dispatch in PCL - Kiczales, Rodriguez - 1990
- 28 Compile-time analysis of object-oriented programs (context) - Vitek, Horspool et al. - 1992
- 22 Optimizing Dynamically-Typed Object-Oriented Languages With .. (context) - Hitzle, Chambers et al. - 1993
- 14 Taming Message Passing: Efficient method lookup for dynamica.. - Vitek, Horspool - 1994
- 9 Compact Dispatch Tables for Dynamically Typed Programming La.. - Vitek - 1995
- 1 Fast Constant Time Type Inclusion Testing (context) - Vitek, Horspool - 1995



The graph only includes citing articles where the year of publication is known.

Documents on the same site (<http://www.csr.uvic.ca/~nigelh/pubs.html>): [More](#)

Incremental Generation of LR Parsers - Horspool (1989) [\(Correct\)](#)

Faster Generalized LR Parsing - Aycock, Horspool (1999) [\(Correct\)](#)

JAZZ: An Efficient Compressed Format for Java Archive Files - Bradley, Horspool, Vitek (1998) [\(Correct\)](#)

[Online articles have much greater impact](#) [More about CiteSeer](#) [Add search form to your site](#) [Submit documents](#)
[Feedback](#)

CiteSeer - Copyright [NEC](#) and [IST](#)

Searching for PHRASE **virtual table class loading**.

Restrict to: [Header](#) [Title](#) Order by: [Expected citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Google \(CiteSeer\)](#) [Google \(Web\)](#)
[CSB](#) [DBLP](#)

No documents match Boolean query. Trying non-Boolean relevance query.

500 documents found. Order: relevance to query.

[System Support for OpenGL Direct Rendering - Kilgard, Blythe, Hohn \(1995\) \(Correct\) \(5 citations\)](#)
 workstations supports direct rendering using **virtualizable** graphics hardware in conjunction with the indirectly. 2.1 Direct Rendering Benefits **Table 1** breaks down the overhead of indirect rendering such as screen real estate. There are three **classes** of contention that **virtualized** graphics for a reality.sgi.com/mjk/direct.ps

[The Virtual Participant: Lessons to be Learned from a Case-Based ... - Masterton \(1997\) \(Correct\) \(3 citations\)](#)
 The **Virtual** Participant: Lessons to be Learned from a the presentation of cases a summary is given (see **Table 1**)The approaches could use the same set of Figure 2) consists of 4 types of objects used to **classify** previous years discussions: The basic messages kmi.open.ac.uk/techreports/papers/kmi-tr-47.ps.gz

[A Model for Transparent Distribution using Java - Milton \(1997\) \(Correct\)](#)
 to garbage collection. Keywords Java, Java **virtual** machine, **virtual** machine, distribution, whether the Sun JVM does this. Dynamic dispatch **tables** similar to C8] are used, and pointers to safe and dynamic. The Java compiler will produce a **.class** file for each Java **class**, which contains all of www.sd.monash.edu.au/research/publications/1997/TR97-14.ps

[Ergodic properties of Anosov maps with rectangular holes - Chernov, Markarian \(1995\) \(Correct\) \(2 citations\)](#)
 Pianigiani and Yorke [22]Imagine a Sinai billiard **table** (with dispersing boundary) in which the dynamics and fractal properties of those measures. AMS **classification** numbers: 58F12, 58F15, 58F11 Keywords: mpej.unige.ch/mp_arc/html/mp_arc/c/97/97-264.ps.gz

[Why Aren't Operating Systems Getting Faster As Fast as Hardware? - Ousterhout \(1990\) \(Correct\) \(158 citations\)](#)
 However, many parts of the kernel, including the **virtual** memory system and interprocess communication, for hardware and software designers. 2. Hardware **Table 1** lists the ten hardware configurations used for not improve dramatically in future machines, some **classes** of applications may be limited by memory www.cs.berkeley.edu/projects/sprite/papers/os-benchmark.ps

[A Comparative Study of Inductive Logic Programming Methods.. - Cohen, Devanbu \(1997\) \(Correct\) \(2 citations\)](#)
 hierarchy. These relations, described in **Table 1**, were then extracted from the source code using (ILP) methods for predicting fault density in Cclasses. In this problem, each training example is a www.research.att.com/~prem/ml97.ps

[Learning Dexterous Manipulation Skills for Multifingered Robot.. - Fuentes, Nelson \(1996\) \(Correct\)](#)
 Arbib et al. 1] introduced the concept of **virtual** fingers as a model for task representation at The goal of the learning system is to build a **table**, indexed by goals and objects that gives as ftp.cs.rochester.edu/pub/u/nelson/1996_us_japan.ps.gz

[Pvm 3 User's Guide And Reference Manual - Geist, Beguelin, Dongarra. \(1994\) \(Correct\) \(308 citations\)](#)
 installed and used. PVM stands for Parallel **Virtual** Machine. It is a software package that allows a : 44 10.2.2 Host **Table** : www4.informatik.uni-erlangen.de/~tsthie/Papers/pvm-ug.ps.gz

[Fast Prolog with an Extended General Purpose Architecture - Holmer, Sano, Carlton. \(1994\) \(Correct\) \(6 citations\)](#)
 ramifications of this on the word format and the **virtual** memory system. Then we present the interface cache data interface cache instruction **table** segment on chip 26 Figure 1: Block Diagram of the pipeline. Hardware interlocks are provided for both **load** and store delays. If data from a **load** instruction www.info.ucl.ac.be/people/PVR/bam.ps

[Taking Steps: The Influence of a Walking Technique on.. - Slater, Usoh, Steed \(1995\) \(Correct\) \(7 citations\)](#)
 Influence of a Walking Technique on Presence in **Virtual** Reality 1 Mel Slater, Martin Usoh, Anthony the movement was. The questions are shown in **Table 1**, with results given for both experiments (the and consequences. Whether or not a system can be **classified** as immersive, depends crucially on the www.cs.ucl.ac.uk/staff/ucacmxu/Papers/tochi.ps.gz

Higher Harmonics Beam Loading Compensation For A Broad Band... - Saito Fujieda (1998) (Correct)
 the plates of high-power tubes such as 150 -300 kW **class** tetrodes. From the viewpoint of the scale of the
 Higher Harmonics Beam **Loading** Compensation For A Broad Band Ma-Loaded Rf
 Beam **Loading** Compensation For A Broad Band Ma-Loaded Rf Cavity K. Saito 1 ,M. Fujieda, Y. Mori, H.
 hadron.tanashi.kek.jp/jh/apac98/5D010.ps

OODB Bulk Loading Revisited: The Partitioned-List Approach - Wiener, Naughton (1995) (Correct) (7 citations)
 B tree The basic algorithm requires enough **virtual** memory to store the id map. However, it accesses
 map is stored as an in-memory open addressing hash **table** hashed on surrogate. It is kept separate from the
 data quickly. We categorize data sets into three **classes** of sizes, relative to the amount of physical
 www-db.stanford.edu/~wiener/papers/partitioned-list.ps

Prediction of Crack Growth under Variable-Amplitude Loading in... - Newman Jr (1997) (Correct)
 of Crack Growth under Variable-Amplitude **Loading** in Thin-Sheet 2024-T3 Aluminum Alloys J. C.
 model to study fatigue crack growth under various **load** histories. The model was based on the Dugdale
 crack growth rates under constant-amplitude **loading** and then used to predict crack growth under
 techreports.larc.nasa.gov/pub/techreports/larc/1997/mtg/NASA-97-iceaf-jcn.ps.Z

Dynamic Registration Correction in Video-Based Augmented... - Bajura, Neumann (1995) (Correct) (15 citations)
 uneumann@cs.usc.edu KEYWORDS: Augmented Reality, **Virtual** Reality, Registration. ABSTRACT This paper
 usc.edu/pub/graphics/papers/cga.ps

Persistence and Migration for C++ Objects - Shapiro, Gautron, Mosseri (1989) (Correct) (30 citations)
 operating system techniques, such as paged **virtual** memory, untyped storage, and demand **loading**,
 allocating an entry for it in the object descriptor **table**. Thus, inheritance is enough to implement easy
 intervention. The key elements are dynamic **classes**, a generalized pointer type which allows to
 lip.inria.fr/INRIA/Projects/SOR/papers/1989/PrsC++_ecoop89.ps.gz

Visual Cues for Perceiving Distances - From Objects To (Correct)
 on the perception of the distance between a fixed **virtual table** and an approaching block in a **virtual**
 perception of the distance between a fixed **virtual table** and an approaching block in a **virtual**
 www.cs.utah.edu/npr/papers/Hu_cues.pdf

TupleRank: Ranking Discovered Content in Virtual Databases - Berlin, Motro (2002) (Correct)
 TupleRank: Ranking Discovered Content in **Virtual** Databases Jacob Berlin Amihai Motro Information
 Consequently, the rows in a discovered **virtual table** have mixed assurance levels, with some rows being
 Typical data integration systems may be **classified** at best as semi-automatic [4, 6, 8, 9]
 www.isc.gmu.edu/techrep/2002/02_03.pdf

Dynamic Optimistic Interprocedural Analysis: a Framework... - Pechtchanski, Sarkar (2001) (Correct) (1 citation)
 are implemented as part of the IBM Jalapeño Java **Virtual** Machine. Our experimental results for the
 Rule[is the set of rules de ned in Appendix A (Tables 7-11)These rules describe the impact of each
 is designed to be used in the context of dynamic **class loading** and dynamic compilation, and includes
 cs1.cs.nyu.edu/~pechtcha/pubs/coopsia01.ps

The Spotless System: Implementing a JavaTM System for... - Antero Taivalsaari Bill (1999) (Correct)
 the Spotless system, which is based on a new Java **virtual** machine developed at Sun Labs and targeted
 efficient internal representations (such as method **tables** and field **tables** to perform efficient runtime
 supporting the full bytecode set and dynamic **class loading**. In this report we describe the design
 www.sun.com/research/techrep/1999/sml_tr-99-73.ps

A Case For Sealing Classes In Java - Biberstein, Sreedhar, Zaks (2002) (Correct)
loading to seamlessly migrate objects between **virtual** machines without having to distribute the
 of **virtual** dispatch **load** the **virtual** dispatch **table** (also called vtable)obtain the appropriate
 A Case For Sealing **Classes** In Java Marina Biberstein Vugranam C.
 www.haifa.il.ibm.com/info/ple/papers/class.pdf

Documents 61 to 80 Previous 20 Next 20

Try your query at: [Google \(CiteSeer\)](#) [Google \(Web\)](#) [CSB](#) [DBLP](#)

CiteSeer - Copyright [NEC](#) and [ISI](#)



Subscribe Register Login
(Full Service) (Limited Service, Free)

Search: ☒ The ACM Digital Library ☐ The Guide

Dixon <and> Method dispatcher

THE ACM DIGITAL LIBRARY

Feedback

Terms used Dixon and Method dispatcher

Sort results
by

relevance



Display results

expanded form



☒ Save results to a Binder

☒ Search Tips

☐ Open results in a new window

Try
Try

Results 1 - 20 of 200

Result page: **1** 2 3 4 5 6 7 8 9

Best 200 shown

1 Minimizing row displacement dispatch tables

Karel Driesen, Urs Hölzle

October 1995 ACM SIGPLAN Notices , Proceedings of the tenth annual conference systems, languages, and applications, Volume 30 Issue 10

Full text available: pdf(1.81 MB)

Additional Information: full citation, abstract, references, ci

Row displacement dispatch tables implement message dispatching for dynamic overhead of one memory indirection plus an equality test. The technique is simple which is, however, restricted to statically typed languages like C++. We show how to use dispatch tables to approximate the same size as virtual function tables. To handle multiple inheritance. Experiments on a number of languages ...

2 A fast method dispatcher for compiled languages with multiple inheritance

R. Dixon, T. McKee, M. Vaughan, P. Schweizer

September 1989 ACM SIGPLAN Notices , Conference proceedings on Object-oriented applications, Volume 24 Issue 10

Full text available: pdf(460.71 KB)


Additional Information: full citation, abstract, references,

This paper addresses the problem of an efficient dispatch mechanism in an object-oriented language with multiple inheritance. The solution suggested is a direct table indexed branch such as is used in C++ where assignments are made using a coloring algorithm. The method is applicable to C++ (with multiple inheritance added) and Eiffel, and in a slightly slower form to Objective C.

3 Selector table indexing & sparse arrays

Karel Driesen

October 1993 ACM SIGPLAN Notices , Proceedings of the eighth annual conference systems, languages, and applications, Volume 28 Issue 10

Full text available:  pdf(1.30 MB)

Additional Information: full citation, references, citings

4 A model for implementing an object-oriented design without language exten

Jennifer Hamilton

January 1996

ACM SIGPLAN Notices, Volume 31 Issue 1

Full text available:  pdf(939.62 KB)

Additional Information: full citation, abstract,

This paper proposes a means of implementing an object-oriented design in pr directly support the object-oriented paradigm, without requiring language ext hiding, dynamic binding, polymorphism and single inheritance through a type Smalltalk. Effecient dynamic method binding is achieved through direct looku incremental graph-colouring algorithm. The met ...

5 Two-directional record layout for multiple inheritance

William Pugh, Grant Weddell

June 1990 ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1990 confere and implementation, Volume 25 Issue 6

Full text available:  pdf(906.27 KB)


Additional Information: full citation, abstract, references,

Much recent work in polymorphic programming languages allows subtyping ar such systems, we would like to extract a field from a record with the same eff subtyping and multiple inheritance. Methods currently used make field extract a significant overall performance slowdown. We describe a record layout algor offset to each field n ...

6 Optimizing method search with lookup caches and incremental coloring

Pascal André, Jean-Claude Royer

October 1992 ACM SIGPLAN Notices , conference proceedings on Object-oriented applications, Volume 27 Issue 10

Full text available:  pdf(1.70 MB)

Additional Information: full citation, references, citings

Keywords: Smalltalk-80, coloring, efficiency, lookup caches, method search, o statistics

7 Incremental algorithms for dispatching in dynamically typed languages

Yoav Zibin, Joseph (Yossi) Gil

January 2003 ACM SIGPLAN Notices , Proceedings of the 30th ACM SIGPLAN-SIGA programming languages, Volume 38 Issue 1

Full text available:  pdf(444.76 KB)

Additional Information: full citation, abstract, referen


A fundamental problem in the implementation of object-oriented languages is *structure*, i.e., support for quick response to dispatching queries combined with hierarchy and the method families. Previous theoretical algorithms tend to be large hidden constant. In contrast, successful practical heuristics, including *Vitables* (CT) [16] designed ...

Keywords: CT, dispatch, dynamic-typing, hierarchy, incremental, message

8 Subtypes vs. where clauses: constraining parametric polymorphism

Mark Day, Robert Gruber, Barbara Liskov, Andrew C. Myers

October 1995 ACM SIGPLAN Notices , Proceedings of the tenth annual conference systems, languages, and applications, Volume 30 Issue 10

Full text available:  pdf(1.56 MB)

Additional Information: full citation, abstract, references, ci

All object-oriented languages provide support for subtype polymorphism, which works for families of related types. There is also a need, however, to write code with no real family relationship. To satisfy this need a programming language must support polymorphism, allowing for types as parameters to routines and types. We show how programming and separate compilation there must ...

9 Fast algorithms for compressed multimethod dispatch table generation

Eric Dujardin, Eric Amiel, Eric Simon

January 1998 ACM Transactions on Programming Languages and Systems (TOPL

Full text available:  pdf(682.21 KB)

Additional Information: full citation, abstract, references,

The efficiency of dynamic dispatch is a major impediment to the adoption of new languages. In this article, we propose a simple multimethod dispatch scheme. This scheme is applicable to any object-oriented language using a method premonotonous property (e.g., as Cecil and Dylan) and guarantees that dynamic dispatch is the latter being a major requirement for so ...

Keywords: dispatch tables, late binding, multimethods, optimization, pole type

10 Fast algorithm for creating space efficient dispatching tables with applicatic

Yoav Zibin, Joseph Yossi Gil

November 2002 ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on programming, systems, languages, and applications, Volume 37 1

Full text available:  pdf(312.23 KB)


Additional Information: full citation, abstract, refer

The dispatching problem can be solved very efficiently in the single-inheritance setting. It is not clear how to extend one such solution to the multiple-inheritance (MI) setting. This paper shows that the space requirement is increased by a small factor of κ ; This factor can be thought of as a function of the topology of the inheritance hierarchy. On a data set of ~35 hierarchies total, we build a dispatching data structure, based on a ...

11 Bidirectional object layout for separate compilation

Andrew C. Myers

October 1995 ACM SIGPLAN Notices , Proceedings of the tenth annual conference on programming, systems, languages, and applications, Volume 30 Issue 10

Full text available:  pdf(1.87 MB)


Additional Information: full citation, abstract, references, ci

Existing schemes for object layout and dispatch in the presence of multiple inheritance waste space and are slower than systems with single inheritance. This paper describes a new scheme for object layout that produces smaller objects and faster methods. It also automatically optimizes particular uses of multiple inheritance. The bidirectional programming language Theta, and ...

12 Special issue on persistent object systems: Tigukat: a uniform behavioral o

M. Tamer Özsu, Randal Peters, Duane Szafron, Boman Irani, Anna Lipka, Adrian

July 1995 The VLDB Journal &mdash; The International Journal on Very Large Databases

Full text available:  pdf(2.78 MB)

Additional Information: full citation, abstract, referen


We describe the TIGUKAT objectbase management system, which is under development at the Database Systems Research at the University of Alberta. TIGUKAT has a number of characteristics that include a purely behavioral semantics and a uniform approach to modeling data, including types, classes, collections, behaviors, and functions, as well as meta-level well-defined behavior. In this way, the model abstraction ...

Keywords: database management, objectbase management, persistent storage

13 Optimizing multi-method dispatch using compressed dispatch tables

Eric Amiel, Olivier Gruber, Eric Simon

October 1994 ACM SIGPLAN Notices , Proceedings of the ninth annual conference systems, language, and applications, Volume 29 Issue 10

Full text available:  pdf(1.83 MB)

Additional Information: full citation, abstract, references, ci

Optimizing method dispatch is a central issue in object-oriented language imp scheme, used for example by C++, is the only implementation of method disp performance. This property is the main asset of dispatch tables and a major re However, the major drawback of dispatch tables is the space they require. Re been studied in the case ...

14 Efficient implementation of Java interfaces: Invokeinterface considered har

Bowen Alpern, Anthony Cocchi, Stephen Fink, David Grove

October 2001 ACM SIGPLAN Notices , Proceedings of the 16th ACM SIGPLAN confe systems, languages, and applications, Volume 36 Issue 11

Full text available:  pdf(779.52 KB)


Additional Information: full citation, abstract, references,

Single superclass inheritance enables simple and efficient table-driven virtual method table dispatch does not handle multiple inheritance and interfaces. Th misimpression that interface method dispatch is inherently inefficient. This pa implementation techniques, Java interfaces need *not* be a source of significant

15 Extraneous factors in the Dixon resultant formulation

Deepak Kapur, Tushar Saxena

July 1997 Proceedings of the 1997 international symposium on Symbolic and alge

Full text available:  pdf(1.06 MB)

Additional Information: full citation, references, citings, index term

16 Sparsity considerations in Dixon resultants

Deepak Kapur, Tushar Saxena

July 1996 Proceedings of the twenty-eighth annual ACM symposium on Theory of

Full text available:  pdf(864.89 KB)

Additional Information: full citation, references, citings, index term

17 Algebraic and geometric reasoning using Dixon resultants

Deepak Kapur, Tushar Saxena, Lu Yang

August 1994 Proceedings of the international symposium on Symbolic and alge

Full text available:  pdf(972.13 KB)

Additional Information: full citation, abstract, references,

Dixon's method for computing multivariate resultants by simultaneously elimi method is found to be quite restrictive because often the Dixon matrix is singi identically yielding no information about solutions for many algebraic and geo method for the case when the Dixon matrix is singular, but satisfies a conditic based on the proposed extension f ...

18 On the efficiency and optimality of Dixon-based resultant methods

Arthur D. Chtcherba, Deepak Kapur

July 2002 Proceedings of the 2002 international symposium on Symbolic and al

Full text available:  pdf(233.57 KB)

Additional Information: full citation, abstract, references,


Structural conditions on polynomial systems are developed for which the Dixon operator computes exact resultants. For cases when this cannot be done, the degree of the operator computed using the Dixon-based methods is typically minimal. A method based on a combination of Sylvester-dialytic and Dixon methods is proposed. The construction often leads to an exact result ...

Keywords: BKK bound, Bezoutians, Dixon method, Sylvester-type matrices, multiplier matrices, resultant, support

19 Efficient multiple and predicated dispatching

Craig Chambers, Weimin Chen

October 1999 ACM SIGPLAN Notices , Proceedings of the 14th ACM SIGPLAN conference on systems, languages, and applications, Volume 34 Issue 10

Full text available:  pdf(2.41 MB)

Additional Information: full citation, abstract, references, citation

The speed of message dispatching is an important issue in the overall performance of a system. We have developed an algorithm for constructing efficient dispatch functions for single dispatching, multiple dispatching, and predicate dispatching. Our algorithm is based on a general predicate dispatching model (which generalizes single dispatching, multiple dispatching, and predicate dispatching) ...

20 Clausal temporal resolution

Michael Fisher, Clare Dixon, Martin Peim

January 2001 ACM Transactions on Computational Logic (TOCL), Volume 21 Issue 1

Full text available:  pdf(277.56 KB)

Additional Information: full citation, abstract, references, citation




In this article, we examine how clausal resolution can be applied to a specific, namely discrete linear temporal logic. Thus, we first define a normal form for arbitrary temporal formulae that can be translated into the normal form, while preserving the complexity of the formulae. Then, we define novel resolution rules that can be applied to formulae in this normal form, proving the correctness and complexity ...

Keywords: resolution, temporal logic, theorem proving

Results 1 - 20 of 200

Result page: **1** 2 3 4 5 6 7

The ACM Portal is published by the Association for Computing Machinery. C

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact](#)Useful downloads:  Adobe Acrobat  QuickTime  Windows Media